

# Bab 1

## Menu-Menu SPSS untuk Persiapan Data

Konsep Menu SPSS pada prinsipnya sama dengan konsep Menu yang ada pada software aplikasi populer lainnya, seperti Excel, Word, PowerPoint, dan sebagainya. Pada software SPSS, ada berbagai menu utama, seperti FILE, EDIT, VIEW, DATA, dan seterusnya. Kemudian pada setiap menu, akan ada submenu, lalu ada sub dari submenu (pilihan), yang biasanya akan diteruskan dengan tampilan kotak dialog yang juga bisa 'berlapis' pengisiannya.

Fungsi Menu SPSS adalah *mempermudah persiapan* perhitungan sebuah metode statistik, tetapi **bukan** sebuah metode statistik tertentu. Sebagai contoh, jika ada sebuah variabel yang berisi data pria dan wanita, sedangkan uji t pada data mensyaratkan *hanya* menghitung karakteristik pria; maka perlu dilakukan proses seleksi data terlebih dahulu, yang bisa dilakukan lewat sebuah menu SPSS. Contoh lain lagi, jika isi sebuah variable, yakni laba usaha akan dikenai pajak 15%; maka penggunaan menu SPSS bisa dilakukan dengan menambah sebuah variabel baru (misal nama LABA\_PJK), tanpa diperlukan proses perhitungan lewat software Excel atau yang lain.

Menu pada SPSS yang khusus melakukan perhitungan statistik adalah menu ANALYZE. Sedangkan menu-menu lainnya berfungsi menyiapkan perhitungan statistik, atau mendukung proses pengolahan data tersebut.

Kata MENU dan FUNGSI sering disebut dalam praktik, yang sebenarnya mengacu pada arti yang sama. Sebenarnya kata FUNGSI lebih baik, karena langsung mengarah pada kegunaan tertentu. Seperti fungsi RANK CASES untuk mengurutkan data, yang lebih tepat digunakan daripada menu RANK CASES. Hal ini sebenarnya sama dengan kata 'data,' 'variabel,' 'sampel,' yang sering digunakan bergantian untuk maksud yang sama. Penggunaan kata 'MENU' hanya disebabkan fungsi-fungsi tersebut terdapat pada Menu di SPSS dan sudah biasa disebut demikian.

## 1.1 Kegunaan Menu

Karena digunakan untuk mengolah data nominal dan ordinal, metode statistik nonparametrik mempunyai ciri khusus, seperti melakukan ranking atau pengurutan data, dan mengubah sebuah variabel dengan prosedur tertentu. Dengan demikian, pada beberapa kasus, bisa saja sebuah data atau variabel harus dilakukan *pengerjaan pendahuluan*, seperti diurutkan dahulu, diubah dengan prosedur tertentu, diseleksi dengan kriteria tertentu, dan sebagainya. Pengerjaan seperti itu akan lebih mudah jika dilakukan lewat beberapa Menu SPSS.

Banyak pengolahan data dengan statistik nonparametrik yang langsung bisa diproses, *tanpa harus* menggunakan lewat persiapan tertentu menggunakan Menu SPSS. Namun, adakalanya data masih dianggap 'mentah' dan perlu dipersiapkan dahulu. Misal data yang didapat adalah variabel Pria saja, sedangkan proses harus menyertakan data Wanita. Maka di sini perlu dilakukan proses penggabungan lewat menu Merger.

Kegunaan lain adalah jika pengguna SPSS akan menjelaskan hasil perhitungan statistik yang didapat. Misal didapat angka korelasi Spearman sebesar 0,86, yang pada output SPSS langsung ditampilkan angka 0,86. Jika angka tersebut akan dijelaskan proses perhitungannya dalam sebuah report tertentu, yang di antaranya data awal harus diurutkan, maka Menu Rank Cases bisa digunakan tanpa harus dilakukan pengurutan secara manual.

Semua Menu SPSS atau submenu SPSS bisa digunakan untuk pengolahan data, namun tidak semua *harus* digunakan. Karena kecuali menu ANALYZE, menu yang lain hanya membantu atau mempersiapkan data untuk diolah lebih lanjut, baik dengan Statistik Parametrik atau Statistik Nonparametrik. Menu yang dibahas adalah Menu DATA dan TRANSFORM, dengan berbagai submenunya (bisa tetap disebut menu):

- Menu/Fungsi COMPUTE, untuk mengolah isi sebuah variabel dengan prosedur tertentu.
- Menu/Fungsi MERGER, untuk menggabung sebuah file atau variabel ke file lainnya.
- Menu/Fungsi RANK CASES, untuk mengurutkan sebuah variabel berdasar kriteria tertentu.
- Menu/Fungsi RECODE, untuk mengubah isi sebuah variabel dengan kriteria tertentu.
- Menu/Fungsi SELECT, untuk melakukan seleksi isi variabel dengan kriteria tertentu.

## 1.2 Menu Compute

Fungsi Compute pada SPSS digunakan untuk melakukan proses perhitungan dengan formula tertentu pada sebuah variabel. Misalkan ada sebuah variabel dengan nama BERAT yang berisi data berat badan 10 orang dan akan dicari orang dengan berat ideal; untuk proses penghitungan BERAT IDEAL, *seluruh* isi variabel BERAT tersebut akan dikurangi 10% dari berat semula. Untuk itu, digunakan fungsi Compute yang dengan formula tertentu akan mengubah isi variabel BERAT semula.

Selain perhitungan matematika, fungsi Compute bisa juga digunakan untuk mengubah variabel BERAT dengan fungsi tertentu, seperti Logaritma, Ln, ABS, dan sebagainya.

Proses Compute pada SPSS *secara otomatis* akan mengubah semua isi sebuah variabel sesuai kriteria yang diberikan. Berkaitan dengan contoh di atas, pengurangan berat sebesar 10% akan mengurangi *semua data (10 orang)* tersebut. Hal ini berbeda dengan software Excel yang bisa memilih sekian sel saja. Dengan ciri seperti ini, sebaiknya hasil dari proses Compute ditempatkan pada variabel dengan nama baru (misal BERAT2), agar jika terjadi kesalahan tidak menghapus data asli.

Dalam praktik statistik nonparametrik, fungsi Compute banyak digunakan untuk proses ranking (mengurutkan) sejumlah data, dan membuat berbagai tabel statistik yang relevan, seperti Chi-Square, z tabel, dan sebagainya.

Berikut disertakan beberapa kasus untuk menjelaskan penggunaan fungsi Compute.

### KASUS 1

Berikut produktivitas pekerja saat membuat roti, jika ia bekerja sendirian dan jika bekerja bersama-sama. (lihat file **sign2**)

Pekerja	Sendiri	Bersama
1	50	51
2	56	55
3	57	58
4	54	56
5	50	49
6	49	50
7	48	50
8	59	58
9	57	56

10	56	55
11	55	57
12	58	59

### Penjelasan Data

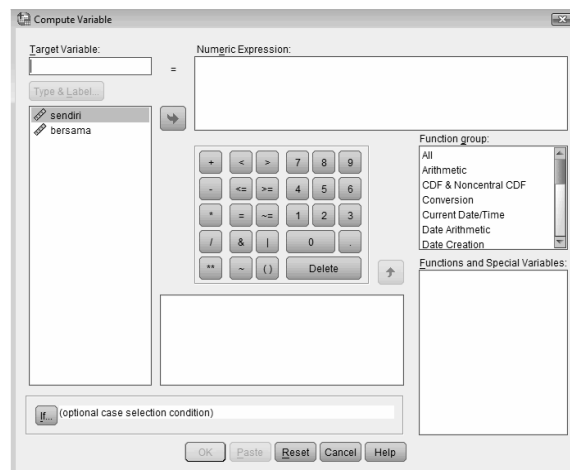
- Input berupa data numerik.
- Data nomor 1 menyatakan bahwa pekerja 1 jika bekerja sendirian menghasilkan roti 50 buah per hari. Namun, jika ia bekerja dengan berkumpul bersama pekerja lain, produktivitasnya menjadi 51 roti per hari. Demikian seterusnya untuk 11 data yang lain.

Dari data di atas didapat temuan, yakni jika seorang pekerja bekerja bersama-sama untuk waktu sebulan lebih, akan terjadi **PENINGKATAN PRODUKTIVITAS 15%** dari sebelumnya. Untuk itu, akan dibuat variabel baru yang menunjukkan peningkatan tersebut.

Di sini akan terjadi penambahan variabel baru berdasar pada data variabel **BERSAMA**.

Langkah:

- Buka file **sign2**.
- Pilih Menu **Transform → Compute Variable...** Tampak di layar:



**Gambar 1.1 Kotak dialog Compute Variable**

Pengisian:

- **TARGET VARIABEL** atau nama variabel baru hasil pengolahan data. Untuk keseragaman, ketik **bersama2**.

- NUMERIC EXPRESSION atau tempat penghitungan (matematis atau menggunakan fungsi tertentu) data. Karena berasal dari variabel BERSAMA dan ditambah 15%, maka ketik **bersama \* 1.15**.

Selain dengan langsung mengetik, bisa juga dengan klik variabel BERSAMA di bagian kiri bawah dan masukkan ke NUMERIC EXPRESSION. Sedangkan tanda \* dan angka 1.15 bisa dilakukan dengan mengklik kode tanda dan angka yang ada di bagian tengah bawah.

Abaikan bagian lain dan tekan OK.

### Output

(disimpan pada file data **COMPUTE.sav**)

NB: perhatikan ekstensi file untuk proses transform adalah sav, dan bukan spo.

Pekerja	Sendiri	Bersama	Bersama2
1	50	51	58,65
2	56	55	63,25
3	57	58	66,70
4	54	56	64,40
5	50	49	56,35
6	49	50	57,50
7	48	50	57,50
8	59	58	66,70
9	57	56	64,40
10	56	55	63,25
11	55	57	65,55
12	58	59	67,85

### Analisis

- Angka pertama dari variabel BERSAMA2 adalah penambahan 15% dari variabel BERSAMA, atau:  
 $51 + (15\% \times 51) = 58,65$   
 bisa juga secara rumus yang ada ditulis pada NUMERIC EXPRESSION:  
 $51 \times 1,15 = 58,65$   
 demikian seterusnya untuk isi variabel BERSAMA2 yang lainnya.

## KASUS 2

Tetap pada file **Compute**.

Permasalahan selanjutnya adalah adanya temuan lain lagi, yakni jika pekerja bekerja sendirian dan telah mampu menghasilkan roti minimal 55 buah, terjadi PENINGKATAN PRODUKTIVITAS rata-rata 10 roti dari sebelumnya.

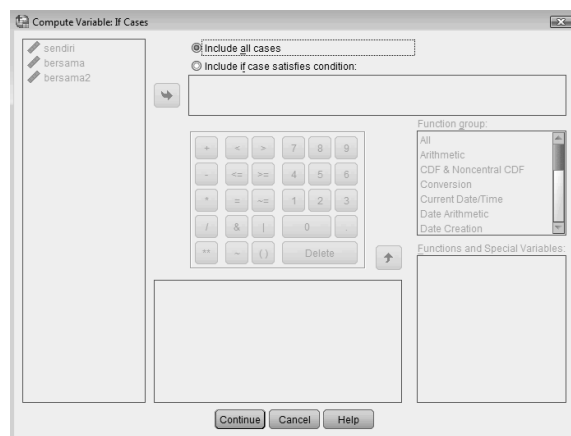
Di sini akan terjadi penambahan variabel baru berdasar pada data variabel SENDIRI.

Langkah:

- Buka file **compute**.
- Menu **Transform** → **Compute Variable...** Tampak di layar kotak dialog COMPUTE.

Pengisian:

- TARGET VARIABEL. Ketik **sendiri2**.
- NUMERIC EXPRESSION. Karena berasal dari variabel SENDIRI dan ditambah 10 buah, maka ketik **sendiri+10**.
- Langkah selanjutnya adalah *menyeleksi data dengan kriteria produktivitas di atas 55*. Untuk itu dari kotak dialog COMPUTE, klik kotak IF, hingga tampak di layar kotak dialog IF.



**Gambar 1.2 Kotak dialog IF**

Pengisian:

Aktifkan pilihan INCLUDE IF CASE SATISFIES CONDITION hingga kotak di bawahnya aktif (tampak warna putih). Kemudian ketik **sendiri>55**.

Kemudian tekan tombol CONTINUE untuk kembali ke kotak dialog utama.

Abaikan bagian lain dan tekan OK.

### Output

(lihat file **COMPUTE2.sav**)

Pekerja	Sendiri	Bersama	Bersama2	Sendiri2
1	50,00	51,00	58,65	,
2	56,00	55,00	63,25	66,00
3	57,00	58,00	66,70	67,00
4	54,00	56,00	64,40	,
5	50,00	49,00	56,35	,
6	49,00	50,00	57,50	,
7	48,00	50,00	57,50	,
8	59,00	58,00	66,70	69,00
9	57,00	56,00	64,40	67,00
10	56,00	55,00	63,25	66,00
11	55,00	57,00	65,55	,
12	58,00	59,00	67,85	68,00

### Analisis

- Angka pertama dari variabel SENDIRI2 tidak ada (kosong) karena data awal dari variabel SENDIRI adalah 50, yang di bawah 55. Sehingga angka 50 tidak diproses lebih lanjut.
- Angka kedua dari variabel SENDIRI adalah 56, yang karena di atas 55 maka diproses lebih lanjut. Proses adalah menambah angka dengan 10:  $56+10=66$  (lihat angka pada Variabel SENDIRI2 pada baris 2).  
Demikian seterusnya untuk isi variabel SENDIRI2 yang lainnya.

## 1.3 Menu Merger

Menu Merger pada SPSS akan menggabungkan isi satu atau beberapa variabel dari sebuah file ke dalam file yang lain. Sebagai contoh, sebuah file bernama BERAT dengan isi dua variabel, yakni B\_AWAL dan B\_IDEAL, akan digabungkan ke file bernama TINGGI yang mempunyai variabel TINGGI\_1. Penggabungan file bisa dilakukan lewat dua cara. Pertama adalah penggabungan secara kolom, hingga otomatis akan ada 3 variabel pada file

TINGGI; cara penggabungan kedua adalah secara baris (cases), yang berciri variabel tetap satu, yakni TINGGI\_1, namun isi kasus bertambah sejumlah data variabel B\_AWAL dan B\_IDEAL.

Merger File akan efektif jika beberapa variabel yang tersebar di beberapa file akan digunakan bersama-sama. Seperti jika variabel BERAT ada di file tertentu, dan variabel TINGGI ada di file lain, padahal kedua variabel tersebut akan dianalisis secara bersama-sama. Maka daripada menginput ulang variabel BERAT, akan lebih efisien jika variabel BERAT tersebut di-merger (gabung) ke variabel TINGGI, kemudian baru dianalisis bersama.

Beberapa kasus berikut akan menjelaskan penggunaan fungsi Merger File, walaupun tetap disarankan memerhatikan kriteria penggabungan, karena SPSS mensyaratkan beberapa kriteria yang cukup kompleks dalam proses penggabungan kolom ataupun baris.

### KASUS 1

Berikut profil dari 11 pekerja PT DUTA MAKMUR bagian pemasaran yang meliputi Nilai Test Calon Karyawan dan Prestasi Karyawan yang bersangkutan. (lihat file **merger**)

Test	Prestasi
78	79
77	75
75	69
79	81
82	83
85	88
86	90
70	74
80	84
69	71
67	70

NB: Input berupa data numerik, dan berupa skor dengan skala 0 sampai 100.

Data dengan dua variabel di atas akan ditambah satu variabel baru untuk tambahan profil, yaitu variabel dengan nama MOTIVASI pekerja untuk kesebelas pekerja tersebut. Variabel MOTIVASI tersebut sudah terdapat pada file lain dengan nama MERGER 2, dengan tampilan lengkap. (lihat file **merger2**)

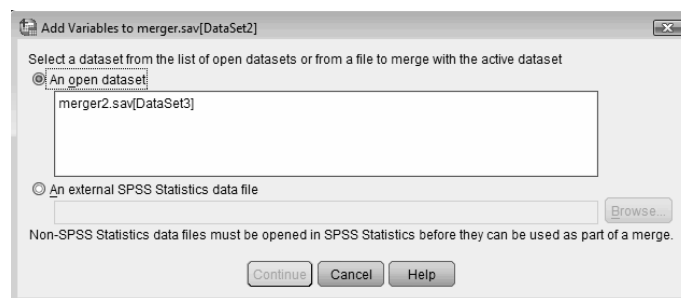


motivasi	absen
84	3
88	2
84	2
82	3
70	1
59	1
59	1
64	4
68	2
91	4
59	4

Perhatikan, pada file MERGER2 terdapat pula variabel motivasi yang akan digabung ke file MERGER.

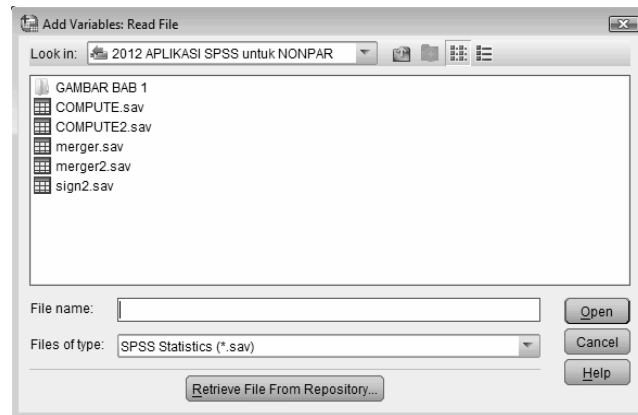
Langkah:

- Buka file **merger**.
- Menu **Data → Merger files → Add variables...** Tampak di layar:



**Gambar 1.3 Kotak dialog Add Variable**

Tekan tombol BROWSE untuk mencari letak file yang datanya akan di-merger. Akan tampak di layar:

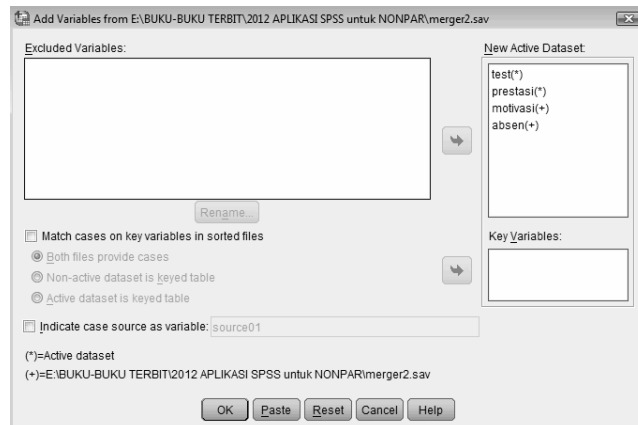


**Gambar 1.4 Kotak dialog Add Variable (2)**

Tampak kotak dialog yang menanyakan nama file yang akan di-merger (digabung) dengan variabel MERGER.

Tentu saja tampilan dapat berbeda-beda, tergantung folder yang ada.

Sesuai kasus, buka Bonus CD Kerja dan pilih file **merger2**. Kemudian tekan OPEN. Lalu setelah tampilan kembali ke kotak dialog awal, tekan CONTINUE untuk melanjutkan proses merger. Tampak di layar:



**Gambar 1.5 Kotak dialog Add Variable (3)**

Perhatikan pada kotak kanan atas yang memuat empat variabel. Hal ini disebabkan file MERGER 2 memuat dua variabel, yakni MOTIVASI dan ABSEN.

Karena hanya dibutuhkan *satu variabel*, yakni variabel MOTIVASI, maka klik mouse pada variabel **Absen** dan pindahkan ke bagian EXCLUDED VARIABLES (dengan klik tanda ◀ yang kini tampak aktif).

Abaikan bagian lain dan tekan OK.

## Output

(lihat file **MERGER VARIABEL.sav**)

NB: output untuk file hasil proses merger adalah berekstensi sav, dan bukan spv.

Test	prestasi	motivasi
78	79	84
77	75	88
75	69	84
79	81	82
82	83	70
85	88	59
86	90	59
70	74	64
80	84	68
69	71	91
67	70	59

Terlihat secara otomatis isi variabel MOTIVASI telah merger (gabung) dengan dua variabel terdahulu dari file MERGER.

Kasus ini adalah contoh proses merger secara kolom, karena terdapat satu tambahan kolom baru tanpa penambahan jumlah kasus (cases).

## KASUS 2

Terkait dengan data MERGER pada kasus sebelumnya. Sekarang file MERGER yang terdiri atas dua variabel (test dan prestasi) akan ditambah Data Baru Pekerja MAKMUR RASA, yakni sebanyak delapan pekerja dengan tetap berisi hasil test dan prestasi pekerja.

Data tambahan tersebut terdapat pada file MERGER 3, dengan tampilan data:

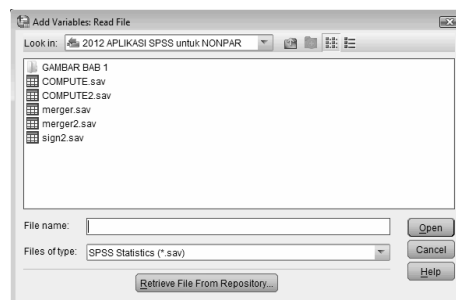
test	Prestasi
75	64
76	65
70	63

69	68
65	70
66	79
70	80
71	88

Langkah:

- Buka file **merger**.
- Menu **Data → Merger files → Add cases...**

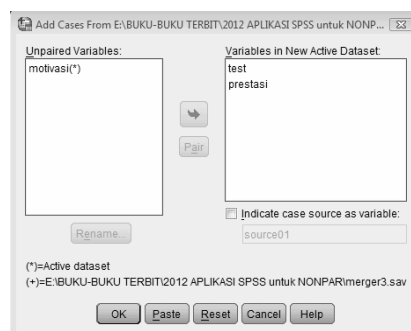
Tampak di layar kotak dialog *Add Cases*. Tekan BROWSE untuk mencari file yang akan ditambahkan kasusnya.



**Gambar 1.6 Kotak dialog Add Cases**

Tampak kotak dialog yang menanyakan nama variabel yang akan di-merger (digabung) dengan variabel MERGER.

Sesuai kasus, buka Bonus CD kerja dan pilih variabel **merger 3**. Kemudian tekan OPEN. Setelah tampilan kembali ke kotak dialog awal, tekan CONTINUE untuk melanjutkan proses merger. Tampak di layar:



**Gambar 1.7 Kotak dialog Add Cases (2)**

Perhatikan pada kotak kanan atas yang memuat dua variabel. Hal ini disebabkan file MERGER 3 memuat dua variabel, yang *mempunyai nama sama dengan variabel pada file MERGER*.

Abaikan bagian lain dan tekan OK.

### Output

(lihat file **MERGER CASES.sav**)

test	Prestasi
78	79
77	75
75	69
79	81
82	83
85	88
86	90
70	74
80	84
69	71
67	70
75	64
76	65
70	63
69	68
65	70
66	79
70	80
71	88

Terlihat otomatis delapan kasus (isi baris) ditambahkan pada file MERGER.

### Tambahan:

Jika pada kotak dialog ADD CASES FROM kotak pilihan **Indicate case source as variable** diaktifkan (klik mouse pada kotak sebelah kiri hingga tanda ✓ tampak), maka tampak kotak kanan bawah (tulisan SOURCE01) menjadi aktif. Ganti kata tersebut dengan **sumber**, dan tekan OK.

Tampak output yang berbeda sedikit dengan output sebelumnya. (lihat file **MERGER CASES sumber.sav**)

Test	Prestasi	Sumber
78	79	0
77	75	0
75	69	0
79	81	0
82	83	0
85	88	0
86	90	0
70	74	0
80	84	0
69	71	0
67	70	0
75	64	1
76	65	1
70	63	1
69	68	1
65	70	1
66	79	1
70	80	1
71	88	1

Terdapat tambahan satu kolom, yakni SUMBER, yang berisi sumber kedua variabel per baris. Kode 0 berarti data awal, sedangkan kode 1 berasal dari file baru yang digabungkan. Jadi, setiap kali terjadi penggabungan secara case (baris), maka jika sumber (source) akan ditampilkan, kode tetap hanya ada dua, yaitu 0 dan 1.

Kasus ini adalah contoh proses merger file secara cases, karena tidak ada tambahan kolom baru; yang ada adalah penambahan baris sebanyak kasus yang ditambahkan.

## 1.4 Menu Rank Cases

Fungsi RANK CASES bertujuan untuk mengurutkan (rank) isi sebuah variabel, yang secara otomatis urutan tersebut akan ditampilkan pada sebuah variabel baru. Sebagai contoh, variabel BERAT dengan isi 170, 174, dan 172, jika diurutkan (ranking) dari terkecil ke terbesar, akan otomatis timbul variabel baru bernama RBERAT dengan isi 1, 3, dan 2.

Proses ranking data (dan juga proses SORT) banyak berperan dalam statistik nonparametrik, karena banyak konsep dasar dari metode statistik nonparametrik adalah mengganti mean (rata-rata) dengan median (titik tengah). Proses mencari median mensyaratkan data harus diurutkan terlebih dahulu, baik dari data terkecil ke data terbesar, atau sebaliknya.

Kasus berikut akan menjelaskan bagaimana proses rank cases digunakan untuk membantu melakukan uji statistik Friedman.

### KASUS 1

File FRIEDMAN berisi data sepuluh orang pekerja MAKMUR RASA yang diberi lima macam perlakuan saat bekerja membuat Roti, dengan hasil produktivitas sebagai berikut. (lihat pada file **friedman**)

pekerja	lampu	musik	televisi	radio	Bonus
ANTON	12	11	21	21	35
ANDY	11	19	25	19	34
ANITA	15	18	28	14	39
ANTONY	12	17	24	15	32
ANIK	15	16	21	16	35
ANTISA	14	18	29	20	28
ANNY	16	19	25	21	21
ANDRE	12	15	22	24	30
AMIR	11	19	26	21	21
AMIN	10	19	24	20	19

Data di atas, untuk keperluan analisis, akan diurutkan (rank) per baris. Sebagai contoh, pekerja bernama Anton, urutan produktivitasnya adalah:

11      12      21      21      35

yang jika diberi angka mulai dari 1 untuk angka terkecil, menjadi:

1      2      3,5      3,5      5

NB: karena angka 21 ada dua, maka dilakukan rata-rata antara urutan 3 dan 4, yakni  $(3+4)/2=3,5$ .

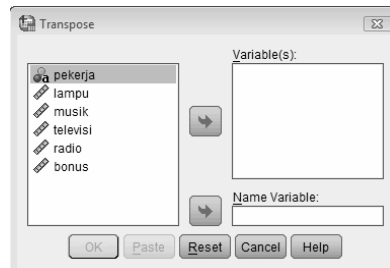
Langkah:

Ada tiga tahapan yang akan dilalui. Pertama adalah melakukan proses *transpose*, kedua mengurutkan hasil transpose tersebut (*rank cases*), dan ketiga adalah melakukan transpose sekali lagi.

#### 1. Proses TRANSPOSE

Pada proses transpose, data akan diurutkan per baris.

- Buka file **friedman**.
- Menu **Data → Transpose...** Tampak di layar:



**Gambar 1.8 Kotak dialog Transpose**

Pengisian:

- VARIABLE(s) atau nama variabel yang akan di-transpose. Karena ini adalah nama variabel yang akan dijadikan baris, masukkan variabel **lampu**, **musik**, **televisi**, **radio**, dan **bonus**.
- NAME VARIABLE. Karena ini adalah variabel yang akan menjadi kolom, masukkan variabel **pekerja**.

Abaikan bagian lain dan tekan OK.

## Output

	CASE_LBL	ANTON	ANDY	ANITA	ANTONY	ANIK	ANTISA	ANWY	ANDRE	AMIR	AMIN
1	lampu	12,00	11,00	15,00	12,00	15,00	14,00	16,00	12,00	11,00	10,00
2	musik	11,00	19,00	18,00	17,00	16,00	18,00	19,00	15,00	19,00	19,00
3	televisi	21,00	25,00	28,00	24,00	21,00	29,00	25,00	22,00	26,00	24,00
4	radio	21,00	19,00	14,00	15,00	16,00	20,00	21,00	24,00	21,00	20,00
5	bonus	35,00	34,00	39,00	32,00	35,00	28,00	21,00	30,00	21,00	19,00

SPSS secara otomatis menampilkan output teks (pada layar lain) berkenaan dengan proses transpose.

Terlihat sekarang ada transpose, di mana isi kolom menjadi baris, dan sebaliknya, isi baris menjadi isi kolom. Karena ada 10 pekerja, maka ada sepuluh kolom pula, ditambah satu kolom tambahan yang *otomatis* dinamakan CASE\_LBL, yang berfungsi untuk menerima nama-nama variabel kolom yang sekarang menjadi baris.

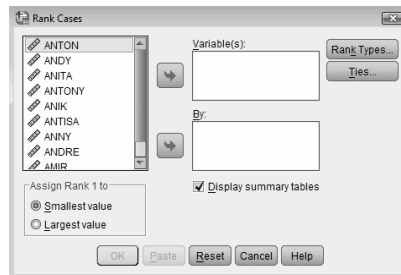
## 2. Proses RANK CASES

Tampilan terakhir tadi akan dilakukan proses pengurutan (ranking), yang otomatis dilakukan per kolom dan menambah sepuluh variabel baru pula.

Proses:

- Menu **Transform → Rank Cases...** Tampak di layar:





**Gambar 1.9 Kotak dialog Rank Cases**

Pengisian:

- VARIABLE(s) atau nama variabel yang akan diranking. Karena pengurutan berdasar kolom, maka masukkan semua variabel nama pekerja, yaitu **anton, andy, anita, antony, anik, antisa, anny, andre, amir, dan amin**.

Abaikan bagian lain dan tekan OK.

Perhatikan isi kotak TIES pada kotak dialog RANK CASES, yang secara default diisi MEAN. Hal ini berarti jika ada sejumlah angka yang sama, otomatis akan diisi rata-rata dari sejumlah angka yang sama tersebut.

Tampilan output teks:

### **RANK**

Created Variables <sup>b</sup>			
Source Variable	Function	New Variable	Label
ANTON <sup>a</sup>	Rank	RANTON	Rank of ANTON
ANDY <sup>a</sup>	Rank	RANDY	Rank of ANDY
ANITA <sup>a</sup>	Rank	RANITA	Rank of ANITA
ANTONY <sup>a</sup>	Rank	RANTONY	Rank of ANTONY
ANIK <sup>a</sup>	Rank	RANIK	Rank of ANIK
ANTISA <sup>a</sup>	Rank	RANTISA	Rank of ANTISA
ANNY <sup>a</sup>	Rank	RANNY	Rank of ANNY
ANDRE <sup>a</sup>	Rank	RANDRE	Rank of ANDRE
AMIR <sup>a</sup>	Rank	RAMIR	Rank of AMIR
AMIN <sup>a</sup>	Rank	RAMIN	Rank of AMIN

a. Ranks are in ascending order.

b. Mean rank of tied values is used for ties.

PERHATIKAN BAHWA OTOMATIS AKAN MUNCUL SEPULUH KOLOM BARU DENGAN ISIAN:

(tampilan di bawah ini DIMULAI DARI KOLOM KESEBELAS, yang bisa dilihat dengan menggeser kolom ke kanan)

RANTON	RANDY	RANITA	RANTONY	RANIK	RANTISA	RANNY	RANDRE	RAMIR	RAMIN
2,000	1,000	2,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000
1,000	2,500	3,000	3,000	2,500	2,000	2,000	2,000	2,000	2,500
3,500	4,000	4,000	4,000	4,000	5,000	5,000	3,000	5,000	5,000
3,500	2,500	1,000	2,000	2,500	3,000	3,500	4,000	3,500	4,000
5,000	5,000	5,000	5,000	5,000	4,000	3,500	5,000	3,500	2,500

Perhatikan tidak semua data hasil rank ditampilkan. Juga setiap kolom dengan variabel baru ditandai dengan huruf 'r' (rank) di depannya. Seperti data untuk pekerja Anton, setelah diranking, isian hasil ranking otomatis dinamakan ranton.

Untuk data dengan nilai sama, maka urutan akan otomatis dibagi secara merata. Sebagai contoh, untuk kolom RANTON, ada angka 3,5 dan 3,5, yang merupakan pembagian dari dua angka yang sama, urutan 3 dan 4, sehingga bernilai:  $(3+4)/2=3,5$ .

Namun untuk kolom RANITA, karena tidak ada angka awal yang sama, otomatis juga tidak ada pembagian nilai urutan.

#### Hapus variabel (data) lama

Langkah selanjutnya adalah menghilangkan sepuluh variabel terdahulu yang berisi angka yang belum diranking, dengan cara:

- Dari tampilan terakhir di atas, blok kolom dengan nama variabel **anton**, **andy**, **anita**, **antony**, **anik**, **antisa**, **anny**, **andre**, **amir** dan **amin**.  
Dengan melakukan blok, semua kolom yang terblok akan berwarna hitam.

Cara blok: klik mouse pada kotak nama variabel, kemudian tetap menekan tombol mouse, blok variabel yang lain.

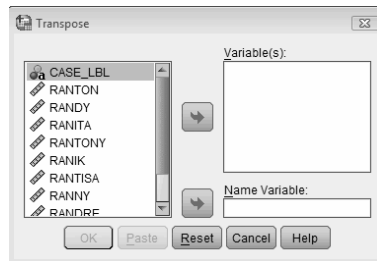
	CASE_LBL	ANTON	ANDY	ANITA	ANTONY	ANIK	ANTISA	ANNY	ANDRE	AMIR	AMIN	RANTON
1	lampu	12,00	11,00	15,00	12,00	15,00	14,00	16,00	12,00	11,00	10,00	2,000
2	musik	11,00	19,00	18,00	17,00	16,00	18,00	19,00	15,00	19,00	19,00	1,000
3	televisi	21,00	25,00	28,00	24,00	21,00	29,00	25,00	22,00	26,00	24,00	3,500
4	radio	21,00	19,00	14,00	15,00	16,00	20,00	21,00	24,00	21,00	20,00	3,500
5	bonus	35,00	34,00	39,00	32,00	35,00	28,00	21,00	30,00	21,00	19,00	5,000

- Klik tombol DEL, maka semua isi yang terblok telah hilang.

#### Proses Transpose sekali lagi

Hasil di atas akan dikembalikan ke posisi semula, yaitu baris menjadi kolom dan sebaliknya.

- Menu **Data** → **Transpose...** Tampak di layar:



**Gambar 1.10 Kotak dialog Transpose**

Pengisian:

- VARIABLE(s) atau nama variabel yang akan di-transpose. Karena ini adalah nama variabel yang akan dijadikan baris, masukkan variabel **ranton, randy, ranita, rantony, ranik, rantisa, ranny, randre, ramir, dan ramin**.
- NAME VARIABLE. Karena ini adalah variabel yang akan menjadi kolom, masukkan variabel **case\_lbl**.

Abaikan bagian lain dan tekan OK.

## Output

Tampak output persis seperti awal kasus, hanya sekarang angka input sebelumnya diganti dengan angka rank (urutan):

(lihat file **friedman-rank.sav**)

	CASE_LBL	lampu	musik	televisi	radio	bonus
1	RANTON	2,00	1,00	3,50	3,50	5,00
2	RANDY	1,00	2,50	4,00	2,50	5,00
3	RANITA	2,00	3,00	4,00	1,00	5,00
4	RANTONY	1,00	3,00	4,00	2,00	5,00
5	RANIK	1,00	2,50	4,00	2,50	5,00
6	RANTISA	1,00	2,00	5,00	3,00	4,00
7	RANNY	1,00	2,00	5,00	3,50	3,50
8	RANDRE	1,00	2,00	3,00	4,00	5,00
9	RAMIR	1,00	2,00	5,00	3,50	3,50
10	RAMIN	1,00	2,50	5,00	4,00	2,50

Perhatikan cara pengurutan yang dilakukan per baris, dan bukannya per kolom.

Output ini sudah siap untuk diproses lebih lanjut dengan metode Friedman.

Tentu saja proses transpose ini akan efektif jika data yang akan di-transpose cukup banyak.

## 1.5 Menu Select Cases

Fungsi SELECT (seleksi) bertujuan untuk memilih sekelompok data pada sebuah variabel yang memenuhi kriteria tertentu. Data yang *tidak terpilih* bisa hanya difilter (dinonaktifkan untuk sementara) atau langsung dihilangkan. Jika data ada yang dihilangkan, yang berarti mengubah isi sebuah variabel awal, sebaiknya variabel dengan data terbaru tersebut disimpan dengan nama lain.

Kriteria untuk memilih sekelompok data tentu tergantung pada sasaran yang akan dituju, yang bisa dibagi ke dalam:

- Jika kondisi tertentu dipenuhi, seperti Berat yang lebih besar dari 500 kilogram, Tinggi Badan yang harus di bawah 170 centimeter, Usia yang ada di antara 20 sampai 30 tahun, dan sebagainya.
- Berdasar nomor kasus tertentu, misal akan diambil kasus (data) nomor 12 sampai nomor 25.
- Berdasar persentase jumlah tertentu, misal akan dipilih 12% dari 100 data Penjualan. Jika demikian, akan diambil 12 data yang dilakukan secara acak oleh SPSS.
- Berdasar kriteria variabel tertentu. Kriteria ini berlaku jika pada satu file terdapat banyak variabel, dan diputuskan sebuah variabel sebagai penyaring (filter) variabel lainnya.

Fungsi Select Cases akan berguna pada jumlah data yang banyak, atau pada file yang berisi banyak variabel, yang pada file atau variabel tersebut akan dihilangkan sebagian datanya untuk proses statistik tertentu. Sebagai contoh, file PROFIL berisi data jenis kelamin pria dan wanita, pendapatan mereka, serta frekuensi beli mereka setiap bulan. Jika sekarang *hanya akan dihitung rata-rata pembelian PRIA yang berpendapatan Rp. 1.000.000,- ke atas*, maka jelas data wanita atau pria namun berpendapatan Rp. 1.000.000,- ke bawah harus dihilangkan (dinonaktifkan) dengan proses SELECT CASES. Jika hal ini tidak dilakukan, maka SPSS akan menghitung *semua data*, yang tentunya akan bias hasil dan tafsirannya.

Selain menyeleksi, bisa juga data yang tidak dipakai dihilangkan atau dinonaktifkan *satu per satu* dengan cara menghapus baris (kasus) di mana data terdapat. Namun, hal ini selain tidak efisien jika data sangat banyak, juga malah bisa timbul kesalahan dalam menghapus data. Proses Select Cases selain cepat dan efisien, juga menjamin ketelitian penyeleksian data, sejauh kriteria yang diberikan tidak salah.

Beberapa kasus berikut menjelaskan penggunaan fungsi SELECT CASES.

## KASUS 1

Berikut profil dari 12 responden yang berhasil dijangkau manajer pemasaran PT MAKMUR RASA untuk mengetahui pendapat mereka tentang roti produk MAKMUR RASA. (lihat file **select**)

Nama	Gender	Usia	Sikap	Income
SULASTRI	Wanita	30,00	Suka	300000,0
BAMBANG	Pria	35,00	Suka	350000,0
RUDI	Pria	24,00	Suka	650000,0
GUNAWAN	Pria	29,00	Cukup Suka	650000,0
BUDIMAN	Pria	26,00	Cukup Suka	245000,0
SRI REJEKI	Wanita	21,00	Suka	400000,0
SUDIRMAN	Pria	40,00	Tidak Suka	750000,0
LILIS	Wanita	26,00	Cukup Suka	540000,0
TANUJAYA	Pria	23,00	Tidak Suka	400000,0
HANDOKO	Pria	35,00	Cukup Suka	650000,0
KIKY	Wanita	31,00	Cukup Suka	550000,0
WIDYAWATI	Wanita		Suka	575000,0
DIDIK	Pria	36,00	Tidak Suka	700000,0
WIWIK	Wanita	35,00	Tidak Suka	675000,0
ANNY	Wanita		Tidak Suka	525000,0

### Penjelasan Data

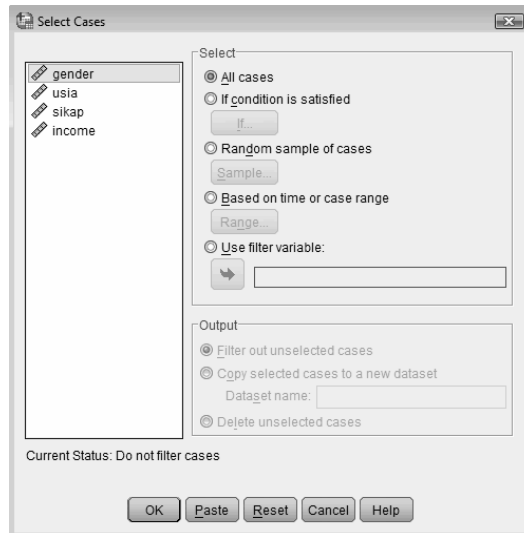
- Variabel NAMA bertipe STRING (bukan berupa angka).
- Variabel GENDER adalah numerik dengan kode 1 untuk Pria dan kode 2 untuk Wanita.
- Variabel SIKAP adalah numerik dengan kode 1 untuk SUKA, 2 untuk CUKUP SUKA, dan 3 untuk TIDAK SUKA.
- Variabel USIA dan INCOME bertipe numerik dan tipe data rasio, sehingga angka 36 untuk Usia ditafsir berusia 36 tahun, dan angka 650000 untuk INCOME ditafsir berpendapatan Rp. 650.000,- per bulan.
- Perhatikan data (baris) 12 dan 15 tidak ada isian untuk variabel USIA, dalam artian usia tidak diketahui atau responden tidak mengisinya.

### Masalah

Manajer Pemasaran hanya memilih gender PRIA saja untuk dilakukan analisis lanjutan, dengan menampung hasil seleksi untuk GENDER PRIA pada file tersendiri.

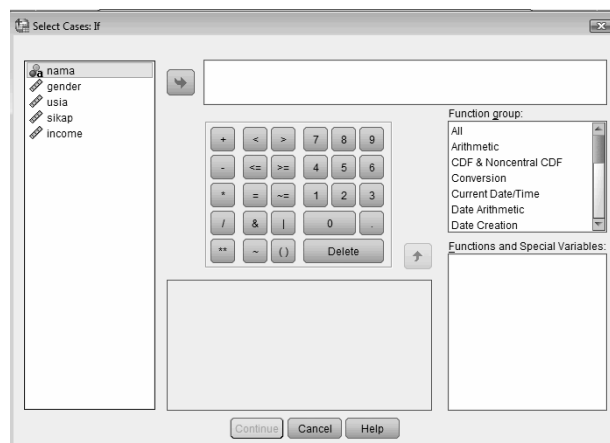
Langkah:

- Buka file **select**.
- Menu **Data → Select Cases...** Tampak di layar:



**Gambar 1.11 Kotak dialog Select Cases**

Karena akan dilakukan select (seleksi data), maka klik mouse pada pilihan **If Condition is Satisfied**, kemudian tekan tombol **IF**. Tampak di layar:



**Gambar 1.12 Kotak dialog Select Cases IF**

Pengisian:

- Karena kode Gender untuk Pria adalah 1, maka ketik pada kotak kanan atas **gender=1**.  
Kemudian tekan CONTINUE untuk kembali ke kotak dialog utama.
- Pada kotak dialog utama (SELECT CASES) tersebut, langkah selanjutnya adalah memilih untuk memberi tanda (filtered) baris atau kasus yang tidak dipilih, ataukah menghilangkan kasus (cases) yang tidak terpilih, dalam hal ini adalah mereka yang mempunyai gender wanita.  
Sesuai kasus, akan dibuat file baru yang khusus pria, berarti akan dilakukan penghilangan gender wanita. Untuk itu, pada bagian OUTPUT (di bagian bawah kotak dialog utama) pilih **Deleted unselected cases**.

Abaikan bagian lain dan tekan OK.

## Output

(lihat file **SELECT 2.sav**)

	nama	gender	usia	sikap	income
1	BAMBANG	Pria	35,00	Suka	350000,00
2	RUDI	Pria	24,00	Suka	650000,00
3	GUNAWAN	Pria	29,00	Cukup Suka	650000,00
4	BUDIMAN	Pria	26,00	Cukup Suka	245000,00
5	SUDIRMAN	Pria	40,00	Tidak Suka	750000,00
6	TANUJAYA	Pria	23,00	Tidak Suka	400000,00
7	HANDOKO	Pria	35,00	Cukup Suka	650000,00
8	DIDIK	Pria	36,00	Tidak Suka	700000,00

Perhatikan semua Gender yang adalah Pria.

Jika pilihan pada bagian OUTPUT adalah **Filtered unselected cases** dan bukannya **Deleted unselected cases**, maka setelah ditekan OK, output tetap menampilkan *semua* isi file semula. Hanya pada sebelah kiri (nomor baris), untuk baris atau kasus yang bergender Wanita ada tanda '/', yang berarti kasus yang bersangkutan tidak terpilih, seperti tampak di bawah. (lihat file **SELECT 2 filter.sav**)

	nama	gender	usia	sikap	income	filter_\$
1	SULASTRI	Wanita	30,00	Suka	300000,00	Not Selected
2	BAMBANG	Pria	35,00	Suka	350000,00	Selected
3	RUDI	Pria	24,00	Suka	650000,00	Selected
4	GUNAWAN	Pria	29,00	Cukup Suka	650000,00	Selected
5	BUDIMAN	Pria	26,00	Cukup Suka	245000,00	Selected
6	SRI REJEKI	Wanita	21,00	Suka	400000,00	Not Selected
7	SUDIRMAN	Pria	40,00	Tidak Suka	750000,00	Selected
8	LILIS	Wanita	26,00	Cukup Suka	540000,00	Not Selected
9	TANUJAYA	Pria	23,00	Tidak Suka	400000,00	Selected
10	HANDOKO	Pria	35,00	Cukup Suka	650000,00	Selected
11	KIKY	Wanita	31,00	Cukup Suka	550000,00	Not Selected
12	WIDYAWATI	Wanita	.	Suka	575000,00	Not Selected
13	DIDIK	Pria	36,00	Tidak Suka	700000,00	Selected
14	WIWIK	Wanita	35,00	Tidak Suka	675000,00	Not Selected
15	ANNY	Wanita	.	Tidak Suka	525000,00	Not Selected

Selain tanda /, secara otomatis SPSS menambah sebuah kolom paling kanan yang berisi data filter kasus, yakni **SELECTED** untuk data yang terseleksi (gender Pria) dan **NOT SELECTED** untuk data yang tidak terseleksi (gender Wanita).

Perbedaan dengan jika di-deleted (hapus), pada pilihan 'Filtered unselected cases' seluruh gender wanita yang telah di'coret' bisa dikembalikan ke file **SELECT** semula dengan cara:

- Dari menu **DATA**, submenu **SELECT CASES** lalu pilihan **ALL CASES**.
- Hapus kolom **FILTER\_\$**; dengan cara: tempatkan pointer pada *kata FILTER\_\$*, lalu blok seluruh isi kolom tersebut, dan tekan **DEL**.

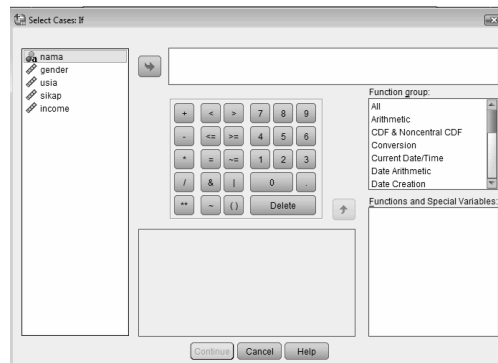
## KASUS 2

Sehubungan dengan kasus **SELECT** sebelumnya, Manajer Pemasaran ingin memilih responden **WANITA** saja. Dan dari responden Wanita yang dipilih, akan diseleksi lagi untuk mereka yang berusia di atas 32 tahun **ATAU** mereka yang mempunyai income di atas Rp. 500.000,- per bulan. Hasil seleksi ini akan disimpan pada file tersendiri.

Langkah:

- Buka file **select**.
- Menu **Data → Select Cases...**  
Tampak di layar kotak dialog **SELECT CASES**. Karena akan dilakukan select (seleksi data), maka klik mouse pada pilihan **If Condition is Satisfied**, kemudian tekan tombol **IF**. Tampak di layar:





**Gambar 1.13 Kotak dialog Select Cases IF**

Pengisian (dibahas secara bertahap):

- Karena kode Gender untuk Wanita adalah 2, maka ketik pada kotak kanan atas **gender=2**.
- Karena disyaratkan usia di atas 32 tahun ke atas, maka ketik **usia>=32**.
- Karena disyaratkan income di atas Rp. 500.000,- maka ketik **income>=500000**.

Tiga persyaratan di atas digabung sehingga menjadi **gender=2 & (usia >=32 | income >= 500000)**.

Ada tanda & untuk menggabung syarat 1 dan 2, sedang tanda | untuk keterangan OR dalam program, yang berarti ATAU. Tanda ( ) menyatakan bahwa usia atau income bersifat 'ATAU'. Jadi, bisa saja usia wanita di bawah 32 tahun, namun income-nya di bawah 500000, atau usia di atas 32 tahun, namun income di atas 500000.

Kemudian tekan CONTINUE untuk kembali ke kotak dialog utama.

- Untuk bagian OUTPUT pilih **Deleted unselected cases**.

Abaikan bagian lain dan tekan OK.

## Output

(lihat file **SELECT 3.sav**)

	nama	gender	usia	sikap	income	filter_\$
1	LILIS	Wanita	26,00	Cukup Suka	540000,00	Selected
2	KIKY	Wanita	31,00	Cukup Suka	550000,00	Selected
3	WIDYAWATI	Wanita	.	Suka	575000,00	Selected
4	WIWIK	Wanita	35,00	Tidak Suka	675000,00	Selected
5	ANNY	Wanita	.	Tidak Suka	525000,00	Selected

Perhatikan semua Gender yang adalah Wanita, yang kemudian diseleksi lebih lanjut (perhatikan tanda ( ) pada penggunaan IF). Karena tanda | atau OR, maka pada output terlihat wanita bernama Lilis walaupun usianya di bawah 32 tahun (26 tahun), tapi karena income-nya di atas 500000 (yakni 540000), maka ia tetap masuk dalam seleksi.

Sedangkan wanita bernama Widyawati dan Anny walaupun usia tidak tertera (tanda ,) namun income juga di atas 500000. Inilah aplikasi dari tanda |.

### KASUS 3

Sehubungan dengan kasus SELECT sebelumnya, karena ada data yang tidak lengkap, Manajer Pemasaran ingin memilih responden yang ada data usianya. Hasil seleksi akan disimpan pada file tersendiri.

Langkah:

- Buka file **select**.
- Menu **Data → Select Cases...**  
Tampak di layar kotak dialog SELECT CASES. Karena akan dilakukan select (seleksi data) berdasar variabel, maka klik mouse pada pilihan **Use filter variable**, kemudian isi kotak yang aktif dengan **usia**.
- Untuk bagian OUTPUT pilih **Deleted unselected cases**.  
Abaikan bagian lain dan tekan OK.

### Output

(lihat file **SELECT 4.sav**)

	nama	gender	usia	sikap	income
1	SULASTRI	Wanita	30,00	Suka	300000,00
2	BAMBANG	Pria	35,00	Suka	350000,00
3	RUDI	Pria	24,00	Suka	650000,00
4	GUNAWAN	Pria	29,00	Cukup Suka	650000,00
5	BUDIMAN	Pria	26,00	Cukup Suka	245000,00
6	SRI REJEKI	Wanita	21,00	Suka	400000,00
7	SUDIRMAN	Pria	40,00	Tidak Suka	750000,00
8	LILIS	Wanita	26,00	Cukup Suka	540000,00
9	TANUJAYA	Pria	23,00	Tidak Suka	400000,00
10	HANDOKO	Pria	35,00	Cukup Suka	650000,00
11	KIKY	Wanita	31,00	Cukup Suka	550000,00
12	DIDIK	Pria	36,00	Tidak Suka	700000,00
13	WIWIK	Wanita	35,00	Tidak Suka	675000,00

Perhatikan pada variabel USIA, di mana sekarang *tidak ada lagi data usia yang tidak tertera*. Dan otomatis dua data dengan usia tidak lengkap akan dihilangkan.

## 1.6 Menu Recode

Fungsi RECODE bertujuan untuk *mengubah* suatu data tertentu menjadi data dengan ciri lain, seperti data yang dulunya bertipe rasio diubah menjadi data bertipe nominal. Sebagai contoh, data rasio berisi Pendapatan seseorang (1.000.000, 2.500.000, 6.000.000, dan sebagainya) akan diubah (recode) menjadi GOLONGAN MISKIN dan GOLONGAN KAYA. Kemudian dua Golongan ini akan diproses lebih lanjut dengan alat statistik nonparametrik tertentu yang mensyaratkan data bertipe nominal.

Pengubahan ini tentu disertai dengan kriteria tertentu, dan ada dua kemungkinan:

- Data hasil proses recode langsung menggantikan data semula, yang disebut RECODE INTO SAME VARIABLES. Jadi -sebagai contoh- angka 1.000.000 *langsung tergantikan oleh GOLONGAN MISKIN*.
- Data hasil proses recode *tidak* menggantikan data semula, namun ditampung pada suatu variabel baru, yang disebut proses RECODE INTO DIFFERENT VARIABLES. Jadi -sebagai contoh- angka 1.000.000 tetap ada, dan timbul variabel baru yang berisi *GOLONGAN MISKIN*.

Fungsi RECODE akan efektif jika data yang akan diubah cukup banyak, serta data sebelumnya bertipe rasio. Fungsi RECODE pada SPSS berkaitan dengan fungsi Transform yang lain, seperti Automatic Recode, Categorize Variables, dan Count (dibahas pada buku lain). Semua fungsi tersebut disediakan SPSS untuk mengantisipasi penggantian variabel atau pengolahan variabel lebih lanjut berdasar pada variabel mula-mula, dengan cara dan kriteria tertentu.

Kasus berikut akan menjelaskan penggunaan Recode yang mengubah data rasio ke data nominal, namun dengan menambah sebuah variabel baru dan tetap menampilkan variabel yang lama.

### KASUS

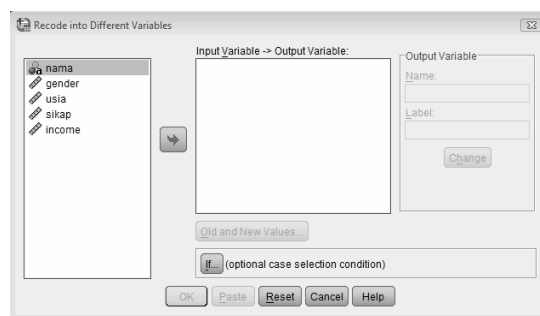
File untuk kasus ini adalah *select* yang sudah dijelaskan datanya pada kasus sebelumnya. Dari data pada file *select* tersebut, khususnya variabel usia, manajer pemasaran ingin mengubah data usia ke data dalam bentuk kode, sehingga bisa diproses untuk uji statistik nonparametrik seperti Chi Square.

Untuk itu, data usia akan dibagi menjadi tiga kelompok:

- Usia sampai dengan 25 tahun digolongkan MUDA.
- Usia > 25 tahun sampai 30 tahun digolongkan DEWASA.
- Usia > 30 tahun digolongkan TUA.

Langkah:

- Buka file **select**.
- Menu **Transform** → **Recode into different variables...** Tampak di layar:

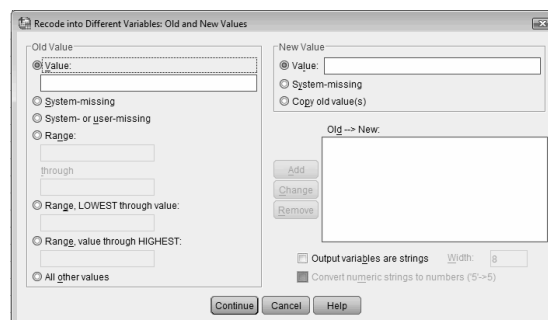


**Gambar 1.14 Kotak dialog Recode**

Pengisian:

- Sesuai kasus, masukkan variabel **usia** ke kotak sebelah kanan.
- Terlihat bagian **OUTPUT VARIABLE** menjadi aktif. Ketik –untuk keseragaman– **golongan** pada bagian **NAME**, kemudian tekan tombol **CHANGE**. Terlihat pada kotak tengah, nama variabel **USIA** akan diganti dengan variabel baru **GOLONGAN**.
- Kemudian tekan tombol **OLD AND NEW VALUES**.

Tampak di layar:

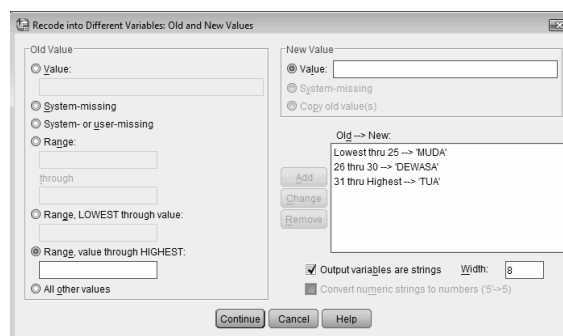


**Gambar 1.15 Kotak dialog Recode**

Proses pengisian:

- Usia sampai dengan 25 tahun menjadi kode MUDA:  
Pada bagian OLD VALUE, pilih RANGE LOWEST THROUGH VALUE dan isi **25**. Hal ini berarti usia terendah sampai 25 tahun.
- Kemudian aktifkan kotak OUTPUT VARIABLES ARE STRINGS (di kanan bawah), dan biarkan isian WIDTH pada angka 8. Hal ini berarti nama kode baru *bisa berupa huruf* dengan karakter maksimal 8 karakter.
- Pada bagian NEW VALUE, pada bagian VALUE, ketik **MUDA**. Hal ini akan menggolongkan mereka yang berusia sampai dengan 25 tahun ke golongan MUDA. Perhatikan proses pengaktifan kotak OUTPUT VARIABLES ARE STRINGS dilakukan terlebih dahulu agar proses ini bisa berjalan.
- Pada bagian kanan tengah, yakni OLD → NEW, terlihat tombol ADD menjadi aktif. Klik tombol ADD, maka *otomatis* terdapat keterangan LOWEST THRU 25 → 'MUDA'.
- Usia di atas 25 tahun sampai 30 tahun menjadi kode DEWASA:  
Pada bagian OLD VALUE, pilih RANGE (empat baris dari atas) dan isi **26** lalu ketik lagi di antara THROUGH angka **30**.  
Pada bagian NEW VALUE, pada bagian VALUE, ketik **DEWASA**.  
Pada bagian kanan tengah, yakni OLD → NEW, klik tombol ADD.
- Usia di atas 30 tahun menjadi kode TUA:  
Pada bagian OLD VALUE, pilih RANGE THROUGH HIGHEST dan isi **31**.  
Pada bagian NEW VALUE, pada bagian VALUE, ketik **TUA**.  
Pada bagian kanan tengah, yakni OLD → NEW, klik tombol ADD.

Jika pengisian benar, tampak di kotak dialog:



**Gambar 1.16** Kotak dialog Recode setelah pengisian kode-kode

Kemudian tekan CONTINUE untuk kembali ke kotak dialog utama.  
 Pada kotak dialog RECODE, abaikan bagian lain dan tekan OK.

## Output

(lihat file **RECODE.sav**)

	nama	gender	usia	sikap	income	golongan
1	SULASTRI	Wanita	30,00	Suka	300000,00	DEWASA
2	BAMBANG	Pria	35,00	Suka	350000,00	TUA
3	RUDI	Pria	24,00	Suka	650000,00	MUDA
4	GUNAWAN	Pria	29,00	Cukup Suka	650000,00	DEWASA
5	BUDIMAN	Pria	26,00	Cukup Suka	245000,00	DEWASA
6	SRI REJEKI	Wanita	21,00	Suka	400000,00	MUDA
7	SUDIRMAN	Pria	40,00	Tidak Suka	750000,00	TUA
8	LILIS	Wanita	26,00	Cukup Suka	540000,00	DEWASA
9	TANUJAYA	Pria	23,00	Tidak Suka	400000,00	MUDA
10	HANDOKO	Pria	35,00	Cukup Suka	650000,00	TUA
11	KIKY	Wanita	31,00	Cukup Suka	550000,00	TUA
12	WIDYAWATI	Wanita	.	Suka	575000,00	
13	DIDIK	Pria	36,00	Tidak Suka	700000,00	TUA
14	WIWIK	Wanita	35,00	Tidak Suka	675000,00	TUA
15	ANNY	Wanita	.	Tidak Suka	525000,00	

Perhatikan secara otomatis SPSS akan mengubah data rasio dari variabel USIA ke berbagai golongan Usia sesuai dengan kriteria yang telah dimasukkan sebelumnya. Dan secara otomatis ada dua kasus, karena data usia tidak ada, otomatis pada variabel GOLONGAN juga kosong.